AppleScript Yardım

Temel bilgiler

AppleScript hakkında

Mac'inizi kullandıkça, bazı görevleri tekrar tekrar yaptığınızı göreceksiniz. AppleScript ile yinelenen görevleri otomatikleştiren betik ve uygulamalar yazabilirsiniz.

AppleScript, OS X'de yerleşik olan güçlü ve çok yönlü bir betik yazma dilidir. AppleScript'i kestirmeler yaratmak, yinelenen görevleri otomatikleştirmek veya size çok fazla zaman kazandıracak özel uygulamalar yazmak için bile kullanabilirsiniz. AppleScript betiğe uygun uygulamaları kontrol etmenizi sağlar.

Bir AppleScript betiği nesneler üzerinde eylemler gerçekleştiren İngilizceye benzer komutlardan oluşur. Bazı eylemlere örnek olarak open (aç), close (kapat), get the name of (şunun adını al) gösterilebilir. Bazı nesnelere örnek olarak bir uygulama, pencere veya tercih ayarları gösterilebilir.

Bir betik bir veya daha fazla uygulamayı kontrol edebilir. Bir uygulamanın betiğe uygun olabilmesi için betiklerinizde kullanabileceğiniz terimleri tanımlaması gerekir. OS X, betiğe uygun birkaç uygulamayla birlikte gelir.

Betikler ayrıca bir ağdaki uygulamaları da kontrol edebilir böylece görevleri birden fazla bilgisayarda otomatikleştirebilirsiniz.

Aslında farkında olmadan AppleScript kullanıyorsunuz. Sistem ve uygulamalardaki birçok işlev aslında uygulama içinden çağrılan AppleScript betikleridir. Örneğin, bu yardım sisteminde uygulama veya dosya açan bir bağlantıya her tıkladığınızda bu eylem gerçekleştiren bir AppleScript betiğidir.

İLGİLİ KONULAR

AppleScript nasıl çalışır
AppleScript Düzenleyici Hakkında
AppleScript dili hakkında
Internet’teki AppleScript kaynakları
AppleScript Studio geliştirme ortamı

Temel bilgiler

AppleScript Düzenleyici Hakkında

AppleScript Düzenleyici, AppleScript betiklerini okuyabilen, yazabilen, kaydedebilen kullanımı kolay bir betik yazma uygulamasıdır.

Bazı AppleScript Düzenleyici özellikleri:

- Birleştirilmiş, kullanımı kolay bir arabirim sunar
- Betiklerideki metni bulmanızı ve değiştirmenizi sağlar
- Uzun betik ifadelerine otomatik kaydırma uygular
- Birden çok geri alma yapmanızı sağlar
- Daha hızlı betik yazma için kestirme menü betikleri sunar

- Uygulama servislerini destekler
- Kitaplık paletini kullanarak uygulama betiklerine erişmenizi sağlar.
- İfadeleri tamamlamanıza yardımcı olmak için Betik Yardımcısı sunar
- Bir betikte neler olduğunu incelemenizi sağlamak için, sonuç tarihçesini ve olay günlüğü tarihçesini sunar.

İLGİLİ KONULAR
AppleScript dili hakkında
AppleScript Studio geliştirme ortamı

Temel bilgiler

AppleScript betikleri yaratma

AppleScript'ten yararlanmak için programlama hakkında birşey bilmenize gerek yok. Aslında, bir kelime betik yazmanıza gerek yok. OS X'de kullanmaya başlayabileceğiniz yüzlerce yararlı AppleScript betiği bulunmaktadır ve web'de de binlercesi mevcuttur.

Biraz daha derine inmek isterseniz, var olan betiklerin ufak değişikliklerle sizin ihtiyaçlarınızı karşılamalarını sağlayabilirsiniz.

Elbette kendi betiklerinizi yaratırsanız daha yararlı olur. Betik yazmayı öğrenmek, işlerinizi daha verimli hale getirecek faydalı araçlar ve zaman kazandıran otomatikleştirmeler yaratmanızı sağlar.

Kendi betiklerinizi yazmanız için basit bir ortam sağlayan AppleScript Düzenleyici'yi kullanarak, kendi betiklerinizi yaratabilir veya değiştirebilirsiniz. AppleScript Yardım'daki tüm örnekler, AppleScript Düzenleyici kullanılarak oluşturulmuştur.

- AppleScript Düzenleyici'yi açın.

  Aşağıdaki metni AppleScript Düzenleyici'ye yazın.

  ```
  say "Welcome to AppleScript!"
  ```

  Çalıştır'ı tıklayın

İLGİLİ KONULAR
AppleScript Düzenleyici Hakkında
AppleScript Studio geliştirme ortamı

Temel bilgiler

AppleScript nasıl çalışır

AppleScript, “Apple olayları” adlı iletiler göndererek çalışır. Bir betiği çalıştırdığınızda, AppleScript Apple olaylarını komutları yerine getiren sisteme veya uygulamalara gönderir.

Pek çok durumda, uygulamalar veya sistem AppleScript'e bir sonuç döndürecektir. Örneğin bir betik, bir uygulamaya bir nesnenin adını almak için bir istek gönderirse, uygulama adı döndürür. Komut çalıştırılamazsa, uygulama veya AppleScript bir hata iletisi döndürebilir. Döndürülen sonuçları AppleScript Editor'da görebilirsiniz.

Genel olarak, Apple olaylarının mekaniğinin farkında olmanız gerekmez. AppleScript bunların tümünü arkaplanda yapar. Sizin yalnızca betiklerinizi yazmanız ve çalıştırmanız gerekir.

İLGİLİ KONULAR

AppleScript hakkında
AppleScript dili hakkında

Temel bilgiler

Uygulamacıklar hakkında

Uygulama olarak kaydedilen bir betik, AppleScript uygulaması veya uygulamacık olarak adlandırılır. Uygulamacıklar, açıldıklarında otomatik olarak çalışır ve betikteki komutları gerçekleştirirler. Uygulamacıklar, birkaç komuttan oluşan küçük işlevler veya binlerce satırdan oluşan karmaşık uygulamalar olabilir.

Uygulamacıklar, düzenlenebilir veya salt çalışır olarak kaydedilebilir. Düzenlenebilir bir uygulamacık, herhangi bir zamanda düzenlenebilmek veya yeniden derlenebilmek için betikleri de içerir. Bir uygulamacığı düzenlemek veya değiştirmek istiyorsanız AppleScript Düzenleyici simgesine sürükleyin. Bir betiği salt çalışır olarak kaydettiğinizde, düzenlenebilir metin bölümü silinir ve dosya artık düzenlenemez veya yeniden derlenemez.

Düzenlenebilir olarak kaydedilen bir eki düzenlemek için AppleScript Düzenleyici veya farklı bir AppleScript düzenleme programında açın. Uygulamacıklar, uygulama açıldığında betiği çalıştırmaya başlayan bir "on run" işleyicisi gerektirir.

- Aşağıdaki örnekte "on run" işleyicisi gösterilmektedir.

```
on run
        --first command
        --second command
end run
```

İLGİLİ KONULAR

Dropletler hakkında
AppleScript dosya türleri

Temel bilgiler

Derleyici betikler hakkında

Derleyici betik, AppleScript Düzenleyici veya farklı bir AppleScript düzenleme uygulmasında okuyup düzenleyebildiğiniz, düzenlenebilir formda bir AppleScript dosyasıdır.

Bir derleyici betiği açtığınızda, AppleScript Düzenleyici'de görünür ancak komutların hiçbirini otomatik olarak gerçekleştirmez. Bu, betiğin çalışması gerekmeden her türlü değişikliği yapmanızı sağlar. Betiği, tüm metni okuyup dosyadaki AppleScript koduna dönüştürecek şekilde derleyebilirsiniz. Derlenmiş betik çalıştırılabilir. Derlenmiş betiğin dosya uzantısı .scpt şeklindedir.

Dosyayı, derlenmiş bir betik olarak kaydettiğinizde metin kodla birlikte dosyada kalır. Bu nedenle, derlenen betiği istediğiniz zaman yeniden düzenleyebilirsiniz.

İLGİLİ KONULAR

Uygulamacıklar hakkında
Dropletler hakkında
AppleScript dosya türleri

Temel bilgiler

Dropletler hakkında

Droplet, klasör veya dosyaları simgelerine sürükleyip bıraktığınızda çalışan bir AppleScript betiğidir. Betik, dosya ve klasörler üzerinde eylemlerini gerçekleştirir.

Uygulamalarda olduğu gibi bir droplet, düzenlenebilir veya salt çalışır olarak kaydedilebilir. Düzenlenebilir bir droplet, herhangi bir zaman düzenlenebilmesi veya yeniden derlenebilmesi için betikleri de içerir. Bir dropleti düzenlemek veya değiştirmek istiyorsanız Betik Düzenleyici simgesine sürükleyin. Bir betiği salt çalışır olarak kaydettiğinizde, düzenlenebilir metin bölümü silinir ve dosya artık düzenlenemez veya yeniden derlenemez.

Dropletler, açıldıklarında betiği çalıştırmaya başlayan bir "on open" işleyici gerektirir. "on open" işleyici, droplete sürüklenen öğelerin listesini içeren bir parametreye sahiptir. Droplet betiğin bir bölümü listeyi inceler diğer bölümü öğelerde üzerinde çalışır. Droplet yalnızca bir öğe üzerinde çalışmak için tasarlanmadığı sürece, her öğede çalışma gerçekleştirme döndüren listedeki öğelerin yinelenmesini sağlayan tekrarlama döngüsü içermelidir.

- "on open" ve parametre listesi olan droplet işleyici örneği:

```
on open (theItemsDropped)
        --first command
        --second command
end open
```

İLGİLİ KONULAR
Uygulamacıklar hakkında
Derleyici betikler hakkında
AppleScript dosya türleri

Temel bilgiler

AppleScript sözlüklerini kullanma

Bir uygulamayı kontrol ederken hangi komutları ve nesneleri kullanabileceğinizi bulmak için ilgili uygulamanın sözlüğüne bakın. Sözlük, bir uygulama tarafından anlaşılan bir terimler kümesinden oluşur.

AppleScript Düzenleyici'de bir uygulamanın sözlüğünü uygulamayı Kitaplık penceresinde seçerek veya uygulamanın simgesini AppleScript Düzenleyici simgesine sürükleyerek görüntüleyebilirsiniz.

Bir sözlükte bulunan terimler, "paket" adı verilen ilgili terim grupları olarak düzenlenir. Komutlar ve nesneler kalın metindir.

---

1. AppleScript Düzenleyici'de, Pencere > Kitaplık'ı seçin.
2. Kitaplık penceresinde uygulamanın adına çift tıklayın.
3. Bir yazılım paketi seçin.

Yazılım paketindeki terimler ve bu terimlerin açıklamaları görüntülenir.

AppleScript dili farklı kategorilerde düzenlenir:

- Paketler: İlgili terim grupları. AppleScript Düzenleyici Sözlüğü'nde paketler, turuncu bir “S” harfiyle belirlenen ilk sütunda görüntülenir.
- Komutlar: bir eylem veya sonuç isteyin. “Olaylar” da denilen komutlar, mavi bir “C” harfiyle belirlenen ikinci sütunda görüntülenir (ayrıca bir olay olarak da belirtilir).
- Sınıflar: Betik yazılabilen nesneler. Sınıf açıklamaları, sınıfta bulunan özellikleri ve öğeleri listeler. Sınıflar, eflatun bir “C” harfiyle belirlenen ikinci sütunda görüntülenir.
- Özellikler: Değer içeren nesneler. Özellikler, eflatun bir “P” harfiyle belirlenen üçüncü sütunda görüntülenir.
- Öğeler: Diğer nesnelerde bulunan nesneler. Örneğin bir paragraf birçok sözcük içerebilir. Öğeler, turuncu bir “E” harfiyle belirlenen üçüncü bir sütunda görüntülenebilir.

İLGİLİ KONULAR

Sınıf tanımları
Komut tanımları
Uygulamaları yönetmek için AppleScript kullanma

AÇMA

AppleScript Dil Kılavuzu: Komut Referansı
AppleScript Dil Kılavuzu: Sınıf Referansı

Temel bilgiler

Sorunları Çözme

AppleScript'le ilgili bir sorununuz varsa, sorunu çözmeye ilişkin bilgiler edinmek için aşağıdaki konulara bakın.

Kullanıcı, Listeden Seç sorgu kutusunda İptal'i tıkladığında bir hata görünüyorsa

Klasör işlemleriniz beklenmedik bir şekilde başlatılırsa

Betiğiniz bir diski tanımazsa

Temel komutlar

Sınıf tanımları

Bir sınıf tanımı betiğe uygun nesnenin özellik ve öğelerini açıklar. Sınıf bir cümledeki isimle ve sınıf tanımı ismin sözlük tanımına benzetilebilir.

Genellikle AppleScript sözlüğündeki bir sınıf tanımı terimle başlar, kalın stildedir ve aynı satırda terimin kısa açıklamasıyla devam eder.

Bu, betiğe uygun nesneyi oluşturan veya ona ait tüm öğe ve nesnelerin listesiyle devam eder. Bir öğe listesi, bazı betiğe uygun nesneler başka betiğe uygun nesneler içerdiğinden ve bazı nesneler de bu tür nesneler içermediğinden tüm sınıf tanımlarına dahil değildir.

Bir sınıf tanımı, betiğe uygun nesne ve onların değerlerinin özellik veya kalite listesiyle sonlanır.

Özellik girişi özelliğin adıyla başlar ve şununla devam eder:

- Tamsayı(sayı), nokta(iki sayının bir listesi), dizgi(metinsel veri), ve Boolean (true veya false) gibi özelliğin depolanan değerinin veri biçiminin açıklaması.
- Özellik değerinin "salt okunur" [r/o] olduğunu gösteren, isteğe bağlı bir tür göstergesi (yalnızca okunur ve değiştirilemez).
- Özelliğin kısa bir açıklaması

Nesne çoğul formdaysa birden fazla nesneye başvurabilirsiniz.

Nesne bir öğe listesine sahipse farklı nesneler içerir. Betiklerinizde kullanabileceğiniz öğe adı kalın stilde görüntülenir. Düz metin öğeye nasıl başvurabileceğinizi açıklar.

Nesneler özelliklere sahip olabilir. Özellik adları kalın stilde görüntülenir. Düz metin, özelliğin içerdiği değerin türünü (örneğin dizgi) açıklar. Özellikler sabir değer veya sabit listesnden biri olabilir. Bir betik , sözdiziminden sonra [s/o] (salt okunur) olduğu sürece bir özelliğin değerini alabilir ve ayarlayabilir.

İLGİLİ KONULAR

Komut tanımları
Nesne tanımları
AppleScript komutları
AppleScript sözlüklerini kullanma

AÇMA

AppleScript Dil Kılavuzu: Sınıf Referansı

Komut tanımları

Komutlar, AppleScript'te eylemleri gerçekleştiren terimlerdir. Bir AppleScript sözlüğünde komutlar orta sütunda görüntülenir ve mavi bir “C” harfiyle belirlenir. Bir komutun tanımı, komutun yaptıkları ve sözdizimiyle ilgili kısa bir açıklama içerir. Sözdizimi, bir betiğe komutu yazdığınız biçimdir. Sözdizimi birkaç bölümden oluşabilir.

Kalın terimler, bir deyimde kullandığınız terimlerdir. Bu terimleri, tam olarak sözlükte göründükleri gibi yazmalısınız.

Mavi metin sözcükleri sizin sağladığınız bilgileri gösterir. Örneğin, tanımladığınız bir nesne veya değişkeni belirleyebilirsiniz.

Komutların bazen isteğe bağlı parçaları olur, bunlar ayraç içinde görünür. Bunları bir deyimde kullanmanız gerekli değildir.

İLGİLİ KONULAR

Sınıf tanımları
AppleScript sözlüklerini kullanma
Olay tanımları
Nesne tanımları

AÇMA

AppleScript Dil Kılavuzu: Komut Referansı

Olay tanımları

Olay tanımı, betiğe uygun bir nesneye hedef olan eylem için kullanılan fiil veya komutu açıklar. Olaylar, nesnelerin yapacakları veya görev alacakları eylem veya fiillerdir.

Genellikle bir olay tanımı terimle başlar, kalın stildedir ve aynı satırda terimin kısa açıklamasıyla devam eder.

Bunu, komutun çalışmasında kullanılan parametre listesi takip eder. İsteğe bağlı parametreler ayraçların içine yerleştirilmiştir, zorunlu parametreler ayraç içinde değildir.

Her parametre parametre adıyla başlar, parametreyle kullanılan veri formatı açıklamasıyla devam eder ve kısa bir parametre açıklamasıyla sonlanır.

Bir komutun tanımı, komutun yaptıklarıyla ve sözdizimiyle ilgili kısa bir açıklama (altı çizili metin) içerir. Sözdizimi, bir betiğe komutu yazdığınız biçimdir. Sözdizimi birkaç bölümden oluşabilir.

Kalın terimler, bir deyimde kullandığınız terimlerdir. Bu terimleri, tam olarak sözlükte göründükleri gibi yazmalısınız.

Düz metin stilindeki kelimeler vermiş olduğunuz bilgiyi gösterir. Örneğin, tanımladığınız bir nesne veya değişkeni belirleyebilirsiniz.

Komutların bazen isteğe bağlı parçaları olur, bunlar ayraç içinde görünür. Bunları bir deyimde kullanmanız gerekli değildir.

Komutlar nesneler üzerinde eyleme geçer. Bir nesnenin tanımı, adının ve kısa bir açıklamasının (altı çizgili metin) dahil olduğu tanımlı nesnenin sözdizimini gösterir. Bir nesnenin sınıfı, örneğin nesnede kullandığınız komutlar gibi, belli nesne özelliklerini belirler.

İLGİLİ KONULAR

Veri tabanı olayları
Sınıf tanımları
Komut tanımları
Nesne tanımları
AppleScript nasıl çalışır

Nesne tanımları

Komutlar nesneler üzerinde eyleme geçer. AppleScript sözlüğündeki bir nesnenin tanımı, adının ve kısa bir açıklamasının (altı çizgili metin) dahil olduğu tanımlı nesnenin sözdizimini gösterir. Bir nesnenin sınıfı, örneğin nesnede kullandığınız komutlar gibi, belli nesne özelliklerini belirler.

Nesne çoğul formdaysa birden fazla nesneye başvurabilirsiniz.

Nesne bir öğe listesine sahipse farklı nesneler içerir. Betiklerinizde kullanabileceğiniz öğe adı kalın stilde görüntülenir. Düz metin öğeye nasıl başvurabileceğinizi açıklar.

Nesneler özelliklere sahip olabilir. Özellik adları kalın stilde ve mavi görüntülenir. Düz mavi metin, özelliğin içerdiği değerin türünü (örneğin dizgi) açıklar. Özellikler sabir değer veya sabit listesnden biri olabilir. Bir betik, sözdiziminden sonra [s/o] (salt okunur) olduğu sürece bir özelliğin değerini alabilir ve ayarlayabilir.

İLGİLİ KONULAR

Uygulamaları yönetmek için AppleScript kullanma
Komut tanımları
Sınıf tanımları
Bir nesneye ne yapacağını anlatma
AppleScript sözlüklerini kullanma

AÇMA

AppleScript Dil Kılavuzu: Sınıf Referansı
AppleScript Dil Kılavuzu: Özellikler

If ifadeleri

If ifadesi, yalnızca belirli koşullar gerçekleştiğinde yürütülen bir komuttur. Her if ifadesi, true veya false olarak değerlendirilen bir Boolean ifadesi içerir. If ifadesinin içerdiği komutlar, Boolean ifadesinin değeri yalnızca true olduğunda yürütülür. If ifadesi end if ile biter. Örneğin:

```
if amountReceived > 100 then
        set memberStatus to "special"
end if
```

Yukarıdaki ifadede, Boolean ifadesi true (doğru) olduğunda (alınan miktar 100'den büyük iken), if ifadesi arasındaki komutlar yürütülür. Boolean ifadesi false (yanlış) ise, komutlar yürütülmez.

If ifadesi else ifadesi ile de oluşturulabilir. Bu birleşim if-then-else ifadesi olarak da bilinir. Örneğin:

```
if amountReceived > 100 then
        set memberStatus to "special"
else
        set memberStatus to "general"
end if
```

Yukarıdaki ifadede, Boolean ifadesi true (doğru) olduğunda (alınan miktar 100'den büyük iken), ifadenin ilk bölümündeki komutlar yürütülür. İlk Boolean ifadesi false (yanlış) ise, ifadenin else'den sonra gelen ikinci bölümündeki komutlar yürütülür.

Birden fazla koşulu ele almak için, if ifadelerini birbirlerinin içine yerleştirebilirsiniz.

AppleScript'teki if ifadeleri hakkında daha fazla bilgi almak için AppleScript Dil Kılavuzu'na bakın.

İLGİLİ KONULAR

Yineleme (Repeat) ifadeleri
Komut tanımları
AppleScript'te değişken ayarlama
Sabit değerler

AÇMA

AppleScript Dil Kılavuzu - If

Yineleme (Repeat) ifadeleri

Koşullara bağlı olarak görevleri birden çok kez gerçekleştirmek için repeat deyimlerini kullanabilirsiniz. Repeat deyimi, bir koşul yerine getirilip döngüden çıkılana kadar bir deyim grubunu döngüde çalıştırmayı sürdürür. Çeşitli durumlara yanıt vermek amacıyla repeat deyimlerini birbiriyle iç içe yerleştirebilirsiniz.

AppleScript, farklı durumlara yanıt verebilen bir repeat deyimleri aralığına sahiptir.

repeat (her zaman)

Sürekli olarak bir döngü içinde yinelenir. Döngüden çıkmak için bir çıkış deyimi kullanın.

```
repeat
        -- statement
        -- statement
        if (x = 5) then exit repeat
end repeat
```

repeat (number) times

Belirtilen sayı kadar tekrarlar.

```
repeat 5 times
        -- statement
        -- statement
end repeat
```

repeat while

Belirtilen bir durum true olduğunda tekrarlar.

```
set x to 1
repeat while (x < 5)
        -- statement
        -- statement
        set x to (x + 1)
end repeat
```

repeat until

Belirtilen bir durum true olana kadar tekrarlar.

```
set x to 1
repeat until (x = 5)
        -- statement
        -- statement
        set x to (x + 1)
```

```
end repeat
```

repeat with (variable) from (startValue) to (stopValue)

Değişkeni startValue değerine ayarlar, ardından her döngüden sonra değişkeni belirli bir miktarda artırır. Değişken stopValue değerine ulaştığında, repeat sona erer.

```
repeat with x from (5) to (25)
        -- statement
        -- statement
end repeat
```

repeat with (variable) in (list)

Değişkeni listedeki ilk öğeye ayarlar. Döngüden her geçildiğinde, değişken bir sonraki öğeye ayarlanır. Listenin tüm öğeleri kullanıldığında, repeat sona erer.

```
set myList to {234, 342, 681, 874}
repeat with x in myList
        -- statement
        -- statement
end repeat
```

AppleScript'teki Repeat deyimleri hakkında daha fazla bilgi edinmek için, AppleScript Dil Kılavuzu'na bakın.

İLGİLİ KONULAR

If ifadeleri
AppleScript'te değişken ayarlama
Sabit değerler

AÇMA

AppleScript Dil Kılavuzu - repeat

Sabit değerler

Sabit, AppleScript'te ayrılmış bir sözcüktür, önceden belirlenmiş bir değer içerir. AppleScript, nesneler hakkında bilgi almak, karşılaştırma yapmak veya aritmetik hesaplamalar yapmak için kullanabileceğiniz çeşitli sabitler içerir. Uygulamalar ve diğer nesneler kendi sabitlerini tanımlayabilir. Bu sabitler, değer sınıfı tanımlarında tanımlanır.

Bir sabitin değerini görmek için, aşağıdaki örneği AppleScript Düzenleyici'ye yapıştırın ve sonra Çalıştır'ı tıklayın. Sabitin değerini görüntülemek için Sonuç öğesini tıklayın.

Sonuç da bir sabittir. Bu, bir komut tarafından döndürülen değeri gösteren bir betik sabitidir.

```
get the current date
```

Şunlar da başka sabit örnekleridir:

- Aritmetik sabitler: pi

- Boolean sabitleri: true, false
- Tarih ve saat sabitleri: minutes, hours, days, weeks
- Metin sabitleri: return, space, tab
- Betik sabitleri: anything, current application, it, me, missing value, my, result

AppleScript'teki sabitler hakkında daha fazla bilgi için, AppleScript Dil Kılavuzu'na bakın.

İLGİLİ KONULAR

AppleScript'te değişken ayarlama
Nesne tanımları
Sınıf tanımları
Olay tanımları

AÇMA

AppleScript Dil Kılavuzu - sabitler

AppleScript'te değişken ayarlama

Değişken, bir değeri saklamak için kullanılan bir betik içinde yaratılan kapsayıcıdır. Bunu yaratırken değişken için tanıtıcı da denilen bir ad girersiniz. Bir betik çalıştırıldığında değişkene bir değer koyabilir, değere erişebilir veya değeri değiştirebilirsiniz. Değişkenler ancak betik çalıştırıldığı sürece vardır. Betik kapandığında, değer kaybolur.

- Bu örnekler benzersiz bir tanıtıcının nasıl belirtileceğini ve bu tanıtıcıya nasıl değer atanacağını gösterir.

```
set myVar to "153"

set x to "Quarterly Reports"

set x to false
```

Değerlerin bir değişkende nasıl saklandığını görmek için, bu örnekleri AppleScript Düzenleyici'ye yapıştırın ve Çalıştır'ı tıklayın. Değişkende saklanan değeri görmek için Sonuç öğesini tıklayın.

AppleScript'teki değişkenler hakkında daha fazla bilgi için, AppleScript Dil Kılavuzu'na bakın.

İLGİLİ KONULAR

Sabit değerler
If ifadeleri
Yineleme (Repeat) ifadeleri

AÇMA

AppleScript Dil Kılavuzu - değişkenler

Bir uygulamanın konumunu belirtme

OS X'deki uygulamalar "paketler" olarak paketlenmiştir. Paket, uygulamayı oluşturan klasör ve dosyaları içeren özel bir dizindir. Bu nedenle, "path" ve "info for" komutları bir uygulama için farklı sonuçlar verebilir.

Bir uygulama için "path to" komutunu kullanırsanız yol genellikle ".app." uzantısıyla biter. Dosya uzantıları Finder'de dosya adının bir parçası olarak görüntülenmeyebilir, ancak yine de bir dosyadaki betiği belirlemek için uzantıları kullanabilirsiniz.

Bir uygulama hakkında bilgi almak için "info for" komutunu kullanırsanız Klasör özelliği genellikle "true" olarak sonuçlanır. Uzantı gizli, ad uzantısı veya görüntülenen ad gibi Sonucun ek özelliklerini kullanarak dosya ve adı hakkında daha fazla bilgi alabilirsiniz.

İLGİLİ KONULAR

path to
info for
Refer to folders, files, and disks
choose application
Finder için betik yazma komutları

Finder komutları

Klasörlere, dosyalara ve disklere gönderme yapma

AppleScript, bazklasör, dosya ve disklere bir yol eklemeden adlarıyla bakmanızı sağlar. AppleScript, belgeler klasörü, masaüstü, ana klasör ve başlangıç diski gibi öğelerin konumunu belirleyebilir.

- Bu örnek, bir betik içinde belirli bir klasörün adını kullanarak ona nasıl gönderme yapılacağını göstermektedir.

```
set x to the path to the desktop folder
```

İLGİLİ KONULAR

Refer to the Finder in scripts
path to
info for
Finder için betik yazma komutları

Finder komutları

Betiklerde Finder'a gönderme yapma

Finder komutlarını kullanmak için bir tell ifadesinde komutları, aynen komutları bir uygulamaya yönlendirir gibi, Finder'a yönlendirirsiniz.

- Betik içinde "tell" ifadesini kullanma örneğini aşağıda görebilirsiniz. Tell ifadesi pencere sayısını "Finder.app." öğesine yönlendirir.

```
tell application "Finder.app"
        return the number of windows
```

```
end tell
```

İLGİLİ KONULAR

Finder
Bir nesneye ne yapacağını anlatma
Refer to folders, files, and disks
Nesne tanımları
Finder için betik yazma komutları

AÇMA

AppleScript Dil Kılavuzu - Nesne belirticileri

Finder için betik yazma komutları

Finder ile kullanabileceğiniz bazı komutlar şunlardır:

activate

alias file

alias list

application file

clean up

close

column

Computer-object

container

copy

count

data size

delete

desktop window

desktop

disk

document file

duplicate

eject

empty trash

exists

file

file information

find

Finder

Finder window

folder

get

icon view options

information window

Internet location file

item

list view options

make

move

moving folder window for

open

package

path to

preferences

preferences window

print

quit

Refer to folders, files, and disks

Betiklerde Finder'a gönderme yapma

reveal

select

set

sort

trash

güncelle

window

Standart betik yazma ekleri komutları

Betik çalıştırma eklentileri betiklerinize birçok faydalı komut ekler.

adding folder items to

beep

choose application

choose color

choose file

choose file name

choose folder

choose from list

choose URL

clipboard info

close access

closing folder window for

current date

delay

dialog reply

display

display dialog

do shell script

download

FTP item

get eof

info for

list disks

list folder

load script

mount volume

offset

open for access

open location

opening folder

path to

POSIX path

random number

read

round

run script

say

scripting components

set eof

set the clipboard to

set volume

store script

summarize

system attribute

time to GMT

upload

URL

write